Teaching AI to Read the Hidden Text That Humans Can See

Originally published Tues 10 Nov 2026 at https://www.unite.ai/teaching-ai-to-read-the-hidden-text-that-humans-can-see/



One of the AI fads that flared up and burned out pretty quickly around 2-3 years ago was using Vision-Language Models (VLMs) to embed cryptic text and 'latent images' into apparently innocuous generated pictures:



Optical illusions and Trompe-l'œil effects characterize the output of the QR Code Monster ControlNet. Source

It's a kind of *Trompe-l'œil* effect-on-demand, and came to maximum prominence via the open source QR Code Monster ControlNet framework.

Once enough people have seen this at work, there isn't really anywhere one can go with the technique, except to use it as a form of practical cryptography, allowing for the sending of text that will not be intercepted or interpreted by VLMs or Optical Character Recognition (OCR) systems – and for the transmission of semi-subliminal messages:



Are you getting the message? Scroll down for the answer, if not. A well-known example of an AI-generated optical illusion, referring (as the main illustration for this article does) to a meme from John Carpenter's 1988 SF outing 'They Live', in which one man discovers that the world is full of subliminal messages from an embedded alien race. Source

Works such as the 2025 study *Hate in Plain Sight*, and last month's *Hidden in Plain Sight* make clear the potential of this method for diffusion of hateful views, and in general, as a method for stenography.

© Martin Anderson and/or commissioning entities

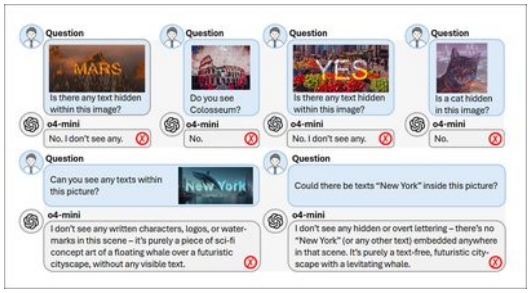
Consequently, it's now regarded in the literature primarily as a potentially malicious technique, rather than an amusing meme-generator.

A Deeper Look

One of the challenges involved in intercepting such incursions – let's call them *AI Opticals*, since no consistent name seems to have emerged – is getting multimodal AI models to 'see the forest for the trees', as it were. When we squint or else reduce an AI Optical to a very small size, the larger characteristics of the subliminal text become easy to see, for us:



Given the gradual way that diffusion model training takes the AI from viewing concepts as 'sketchy roughs' down to being able to render intense details (kind of like finishing the rest of the…owl), VLMs *should* be rather good at discerning patterns of this kind. However, recent research suggests that this is not actually the case:



The 2025 offering 'SemVink' found that an otherwise capable frontier VLM could not penetrate the subliminal text in AI Opticals. Source

However, a recent paper from China suggests a new framework that could help VLMs to recognize AI Opticals.

The new approach, for which code has been released, uses a training method in which an AI model learns from another model, given both a reduced version of the image, and the correct transcription. The researchers report that this enables hidden messages to be read directly from the original images, with substantially higher recognition scores than conventional training methods:

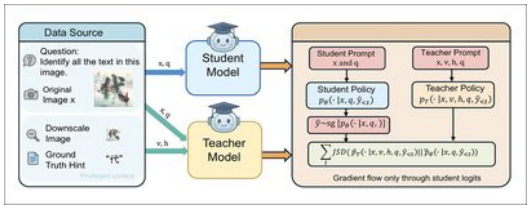


Diagram of the proposed AI training method. Two models are shown learning from the same image. The teacher is given extra clues, including a smaller version that makes the concealed text easier to recognize and the correct answer. Its guidance is used to train the student, which must eventually recognize the text without these clues. Source

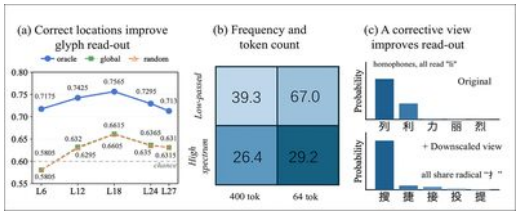
The method, named SQUINT, leverages knowledge distillation, in which information learned by one AI model is passed to another during training. In this case, the student is trained to reproduce the teacher's predictions, using a mathematical comparison of their responses. Once training is complete, the teacher can be discarded, leaving a single model to examine new images without additional assistance.

In SQUINT's two stages, the student's predictions are first compared with those of the better-informed teacher, allowing it to learn from the differences. In the second stage, further improvements are encouraged by rewarding correct readings of hidden text.

In tests, SQUINT recognized hidden lettering in images that had largely defeated the original AI model, also performing better across several independent benchmarks. It was found that hidden text could often be detected *even when it could not be read*, with recognition improving when attention was directed toward the parts of the image that formed the letters.

Further improvements were obtained by suppressing fine details and textures, allowing the underlying shapes to be distinguished more easily:

© Martin Anderson and/or commissioning entities



Three tests investigating why hidden text is misread. Left: directing the model to the correct image regions shows that finding the letters is a major obstacle. Center: removing fine visual detail shows that texture can obscure the larger letter shapes. Right: adding a downscaled view shows that reducing the image helps the model distinguish visually similar characters.

Data and Method

Simply exposing a model to more examples of AI Opticals was found to be insufficient, since conventional training provides no reason for the image to be considered at a different scale. In SQUINT, a downscaled version is therefore used during training to teach the model which visual information should be prioritized when the full-sized image is encountered.

To support this, two datasets were necessary; *SquintBench* is comprised of 1,076 images curated from a (unspecified) production content-moderation pipeline – all drawn from real production data, the authors report, apparently human profile pictures containing hidden text that can be read almost effortlessly by people, with both visible and concealed text transcribed by human annotators.

The second dataset, *SquintCtrl*, features hidden text embedded into generated scenes using QR Code Monster v2 running under the now-venerable Stable Diffusion 1.5 (which apparently must be pried from the cold dead hands of computer vision researchers). Since the text used to generate each image was known in advance, its exact location could be identified automatically, allowing recognition failures to be examined without human annotation.

SquintCtrl was divided into four subsets: *SquintCtrl-Pair* uses 500 matched pairs with and without hidden text to test whether its presence could be detected; *SquintCtrl-Glyph* tested whether individual Chinese and Latin characters could be identified and located; *SquintCtrl-Strength* varied the strength of the hidden text across 200 scenes; and *SquintCtrl-Read* provided 200 open-vocabulary images for testing complete transcription.

Tests

Diagnostic experiments were conducted with Qwen3.5-9B, whose weights were kept unchanged to reveal what the existing model could already perceive.

The first training stage, *Squint OPSD*, uses the better-informed teacher to guide the student. The teacher receives the original image, a downscaled version, and the correct transcription, while the student sees only the original image. Their predictions are subsequently compared, token by token, allowing the student to learn from the teacher while remaining able to work from the original image alone.

Training was conducted on 5,905 prompts, comprising 3,489 synthetic hidden-text images, 1,213 OCR samples and 1,203 visual-question-answering samples. SquintBench was used as the main evaluation set, while transfer was measured on Illusory VQA; AdvOCR; and HC-Bench. Broader vision-language performance was tested on AI2D; DocVQA; and OCRBench.

For comparison, the original pretrained model was tested alongside a version given an additional downscaled view, supervised fine-tuning (SFT), and DAPO. Also evaluated as external references were Claude Opus 5; Gemini 3.1 Pro; GPT-6 Astra; and Qwen3-VL-235B-Thinking.

	SQUINTBENCH	general		
		AI2D	DocVQA	OCRBench
BASE	26.50	86.66	95.93	879
BASE +1/8 view (two-view)	58.96	52.36	95.37	834
SFT (full, single)	72.47	85.52	64.73	722
DAPO	73.55	82.12	92.31	728
SQUINT OPSD	**88.61**	85.33	92.98	833

Test results comparing Squint OPSD with the baseline methods. Squint OPSD achieves the highest SquintBench score, at 88.61, while retaining substantially more general vision-language performance than supervised fine-tuning.

As shown in the results table above, Squint OPSD raised the SquintBench score from 26.50 for the original model to 88.61, outperforming all three training baselines while largely preserving performance on the general vision-language tests.

In the next test, Squint OPSD was evaluated on three external hidden-text benchmarks:

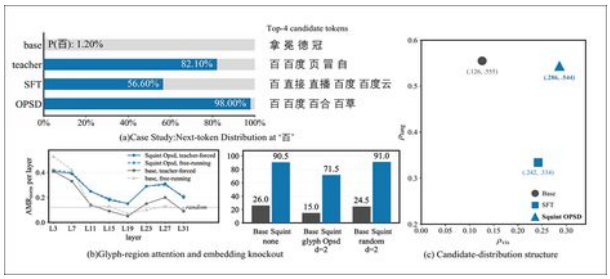
benchmark	cond.	SQUINT OPSD	Qwen3.5-9B base	claude-opus-5	gemini-3.1-pro	gpt-6-astra	qwen3-vl-235b-think
SQUINTBENCH	orig.	**88.61**	26.50	22.26	18.15	8.33	24.44
	1/8 only	**49.64**	38.22	22.41	27.90	8.13	34.37
	40×40 only	**76.12**	68.60	11.07	44.91	6.26	66.88
IIIVQA *Illusion*	WER↓	**67.00**	178.10	166.50	190.13	119.60	177.97
IIIVQA *Filtered*	WER↓	**64.87**	76.80	83.53	70.00	65.27	91.33
AdvOCR†	orig.	**66.7**	3.7	3.7	3.7	11.1	3.7
	1/8 only	**70.4**	44.4	33.3	33.3	0.0	51.9
	40×40 only	**70.4**	51.9	22.2	44.4	44.4	55.6
HC-Bench	orig.	**44.6**	1.8	5.4	5.4	1.79	1.8
	1/8 only	**46.4**	26.8	19.6	33.9	12.5	28.6
	40×40 only	19.6	19.6	8.9	19.6	12.5	**25.0**

© Martin Anderson and/or commissioning entities

Test results comparing Squint OPSD with five other models across SquintBench, and three external hidden-text benchmarks. Squint OPSD performed best in every test when models were given the full, unmodified images. For Illusory VQA, lower word-error rates indicate better performance.

Scores increased from 3.7 to 66.7 on AdvOCR and from 1.8 to 44.6 on HC-Bench, while the word-error rate on Illusory VQA fell from 178.10 to 67.00. Poor results were also obtained by the much larger Qwen3-VL-235B-Thinking and GPT-6 Astra on the original SquintBench images, with scores of 24.44 and 8.33 respectively.

After training, much clearer internal distinctions were observed between the individual characters concealed within an image. For Chinese characters, the measure of how clearly character identity was represented rose from 0.718 to 0.975, while the equivalent score for Latin characters increased from 0.612 to 0.779.



Tests of how the model's handling of hidden characters changed after training. On the left, the model's predictions and attention are shown becoming more concentrated on the image regions that form the concealed characters, with accuracy falling when those regions are disrupted. On the right, the model's alternative character predictions are shown becoming more strongly organized by visual resemblance while retaining language-based information.

Greater attention was also directed toward the parts of an image that *actually formed the hidden letters*. When the model's internal representation of these regions was deliberately disrupted, accuracy fell from 90.5% to 71.5%. By comparison, almost no change was produced by disrupting an equivalent number of randomly chosen background regions, indicating that the concealed text was being read from the relevant visual structures.

Conclusion

Perhaps there is a kind of 'fractal myopia' in this new work that repeats itself across other AI domains. Most users familiar with frontier LLMs and VLMs will recognize the many occasions on which they are prone to concentrate on some small area of text, image or video generation, without broader consideration for how these changes might affect the bigger picture.

In many cases, the issue will be related to the AI's imperfect memory (and its still-nascent ability to compact its own memory, for longer and usually agentic tasks). Sometimes, however, as in this case, it transpires that the default training practices leave something to be desired, and need to be specifically addressed. The question is the extent to which we should learn from such cases and consider revising basic training practice…and when the process can be considered too 'specialized' and marginal for anything but a third-party approach such as LoRA.

First published Tuesday, November 10, 2026

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.
Website: martinanderson.ai
Contact: martin@martinanderson.ai